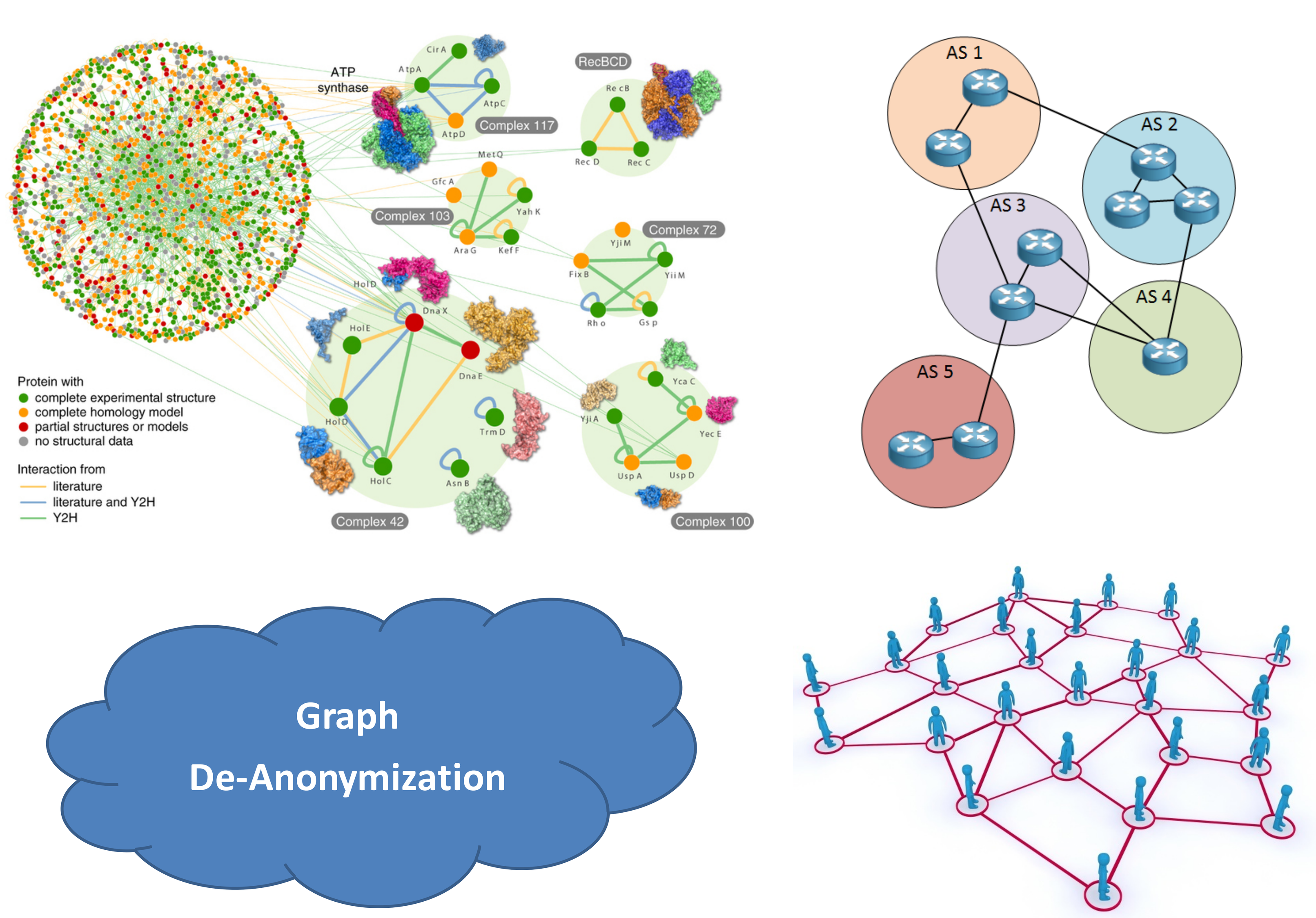
NED: An Inter-Graph Node Metric Based On Edit Distance

Haohan Zhu, Xianrui Meng, George Kollios

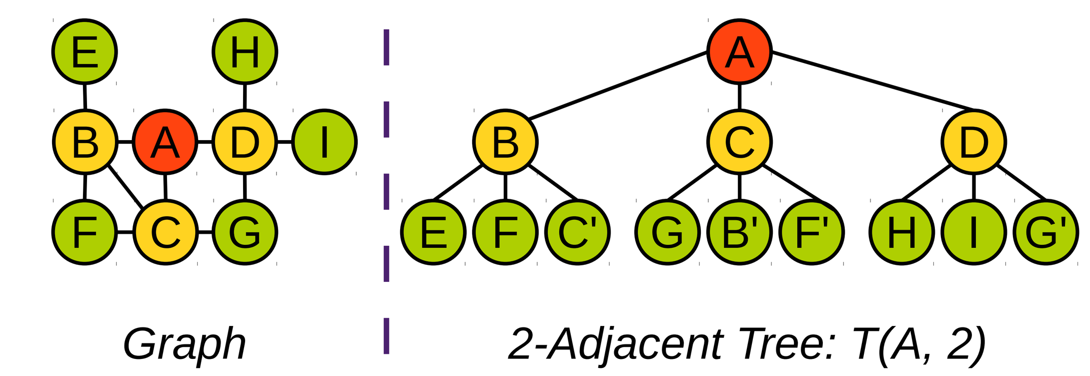
Abstract: Node similarity is a fundamental problem in graph analytics. However, node similarity between nodes in different graphs (inter-graph nodes) has not received a lot of attention yet. The inter-graph node similarity is important in learning a new graph based on the knowledge of an existing graph (transfer learning on graphs) and has applications in biological, communication, and social networks. In this paper, we propose a novel distance function for measuring inter-graph node similarity with edit distance, called NED. In NED, two nodes are compared according to their local neighborhood structures which are represented as unordered k-adjacent trees, without relying on labels or other assumptions. Since the computation problem of tree edit distance on unordered trees is NP-Complete, we propose a modified tree edit distance, called TED*, for comparing neighborhood trees. TED* is a metric distance, as the original tree edit distance, but more importantly, TED* is polynomially computable. As a metric distance, NED admits efficient indexing, provides interpretable results, and shows to perform better than existing approaches on a number of data analysis tasks, including graph de-anonymization. Finally, the efficiency and effectiveness of NED are empirically demonstrated using real-world graphs.

http://arxiv.org/abs/1602.02358

Transfer Learning



Adjacent Tree for Node Similarity



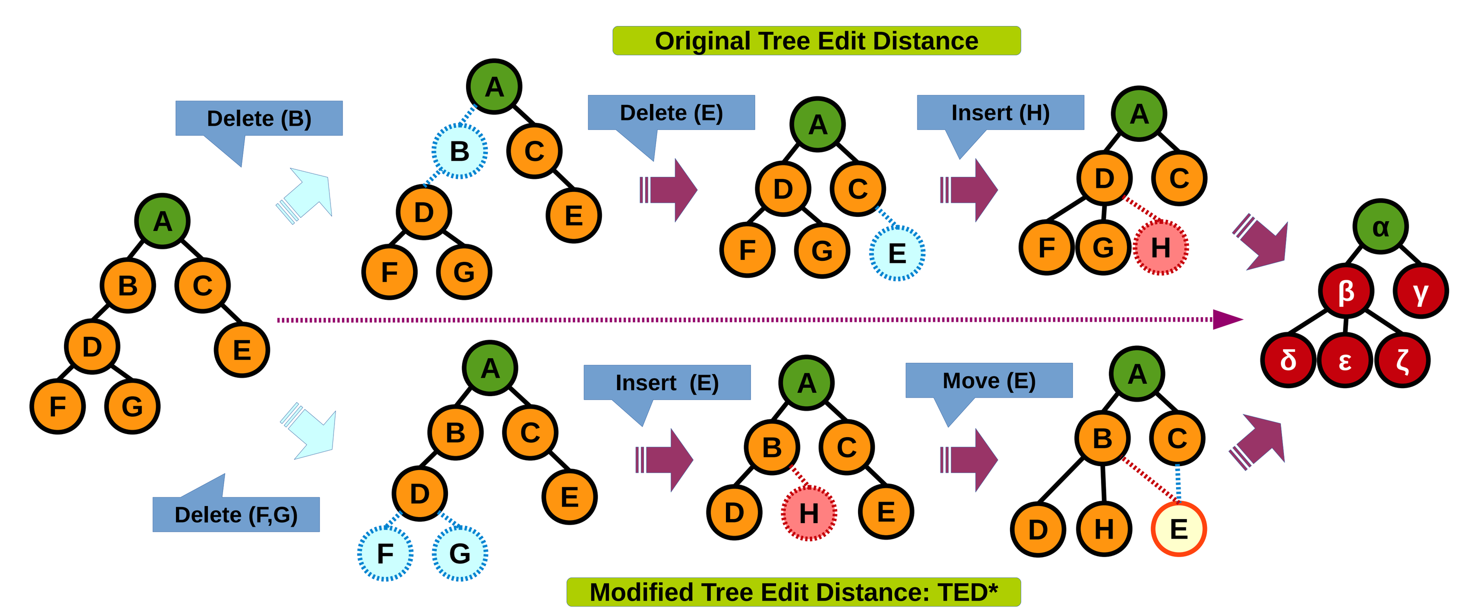
If neighborhood subgraph is used, the node similarity cannot be **METRIC** and **POLYNOMIALLY** computable at the same time.

$$\delta(u,v) = \delta_T(T(u,k),T(v,k))$$

Modified Tree Distance (TED*)

The Computation of classic tree edit distance on unordered unlabeled trees is **NP-COMPLETE** and even **MaxSNP-HARD**

TED* is **POLYNOMIALLY** computable and is also a **METRIC**



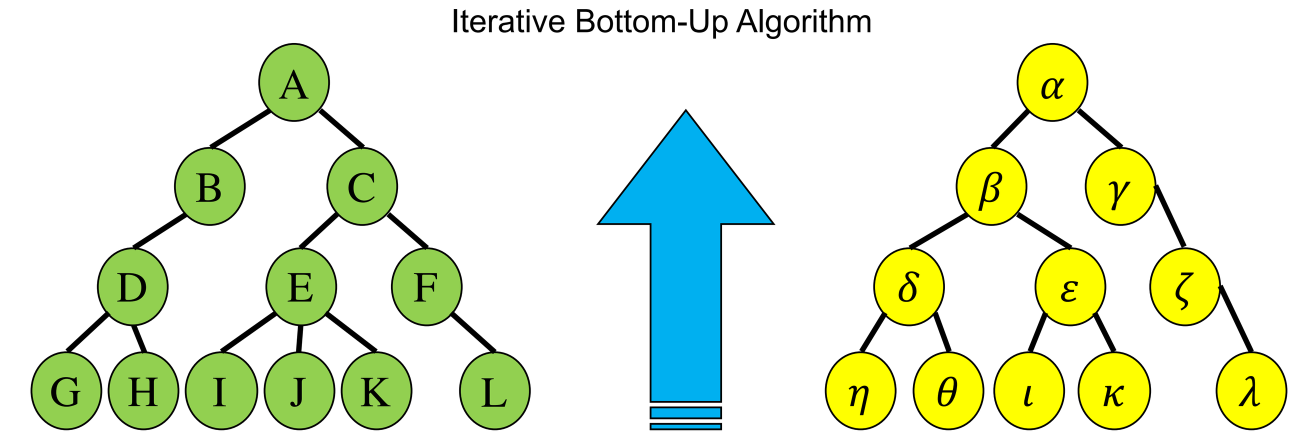
No operation can change the depth of any existing node

Allowed Operations:
① Insert a leaf node;
② Delete a leaf node;
③ Move a node at the same level

$$\delta_T(T(u,k),T(v,k)) \geq 0$$
$$\delta_T(T(u,k),T(v,k)) = \delta_T(T(v,k),T(u,k))$$
$$\delta_T(T(u,k),T(v,k)) = 0, \text{ iff } T(u,k) \simeq T(v,k)$$
$$\delta_T(T(u,k),T(v,k)) \leq \delta_T(T(u,k),T(w,k)) + \delta_T(T(w,k),T(v,k))$$

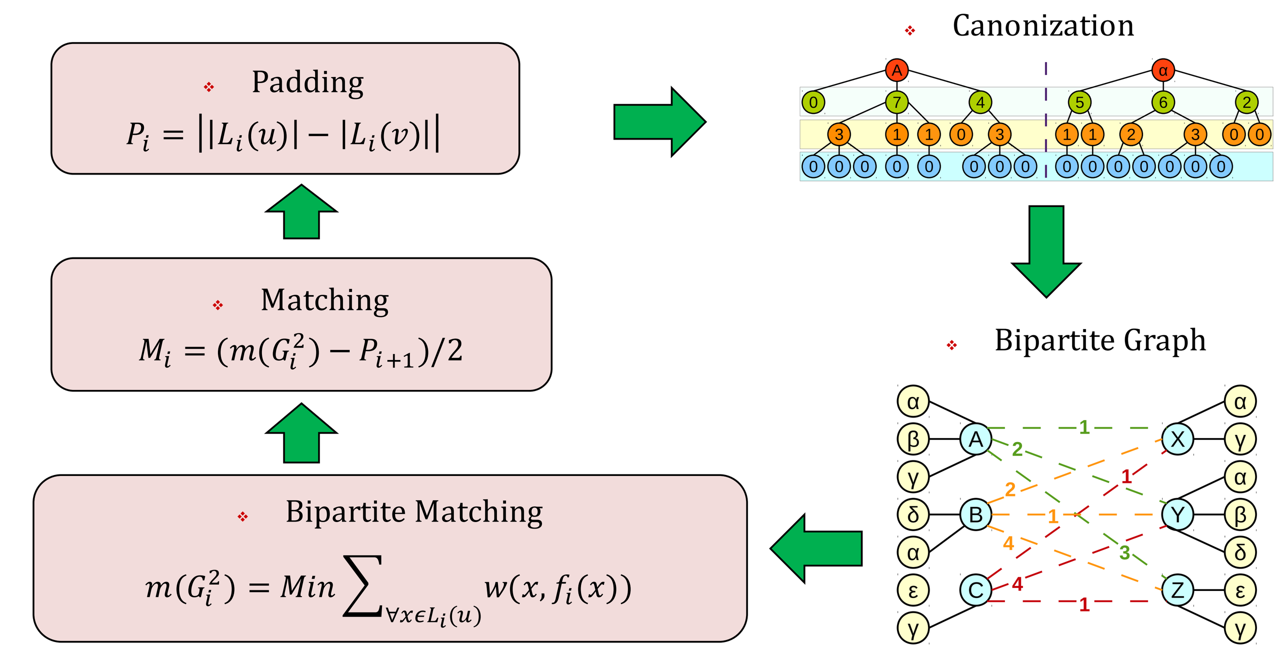
Bottom-Up Alignment

Iterative Bottom-Up Algorithm



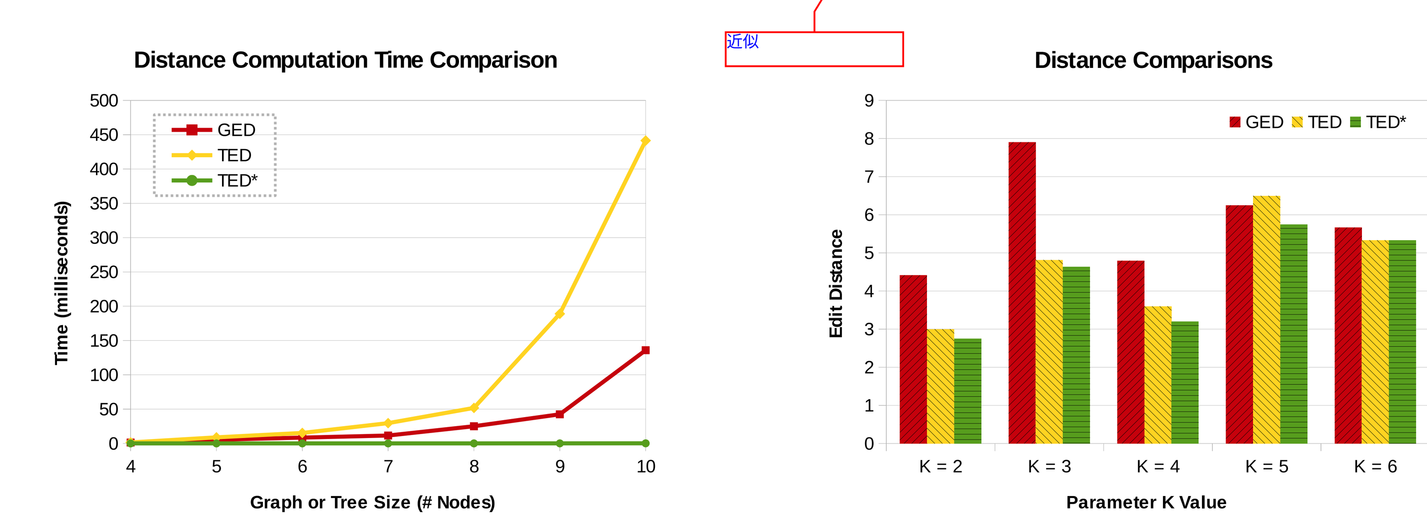
$$\delta_T(T(u,k),T(v,k)) = \sum_{i=0}^{k}(P_i + M_i)$$
$$\delta_T(T(A,3),T(\alpha,3)) = 2$$

Iterative Algorithm



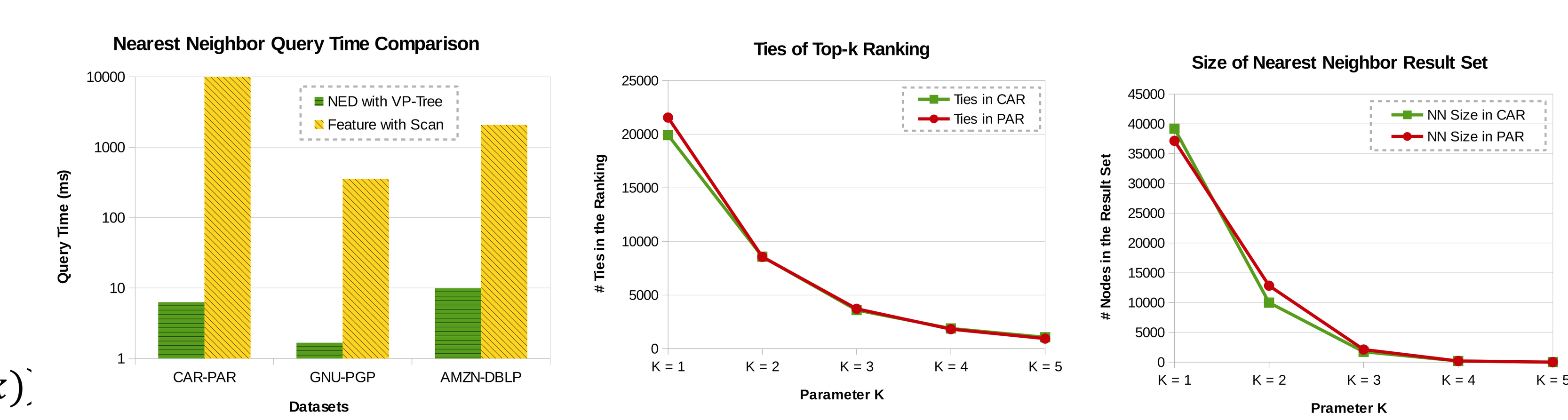
TED* vs TED & GED

A Practical and Good Approximation to Tree Edit Distance



Nearest Neighbor & Top-K Queries

Better Performance and Query Quality



Case Study: Graph De-Anonymization

Higher Precision than Feature-Based Similarities